PROJECT REPORT

Topic: Searching Algorithm

Team member:

21125013 - Vo Hoang Phuc Khang

21125076 - Hoang Thien Duc

21125087 - Vuong Quoc Phong

21125159 - Nguyen Trinh Duy

Contents

I. Introduction:

This is the report for the first group project in the course, which will contain a detailed description of the work we have done so far, the concept and the implementation of algorithms we used in the project, and some observations we inferred during the working process.

II. Work assignments:

Tasks	Assignees	Completion rate
Level 1 Solver BFS	Vo Hoang Phuc Khang	100%
Level 1 Solver DFS	Vo Hoang Phuc Khang	100%
Level 1 Solver UCS	Vuong Quoc Phong	100%
Level 1 Solver A Star	Hoang Thien Duc	100%
Level 2 Solver	Vo Hoang Phuc Khang	100%
Level 3 Solver	Nguyen Trinh Duy	100%
Level 4 Solver	Nguyen Trinh Duy	100%
GUI displayer for algorithm	Vuong Quoc Phong	100%
Exporting Heatmap	Vuong Quoc Phong	100%
Generate test cases	Hoang Thien Duc	100%
Demo video	Nguyen Trinh Duy	100%
Report Documenting	All team members	100%

III. Detailed description

1. Level 1 Solver BFS (Vo Hoang Phuc Khang):
    - Implement the Breadth-First Search (BFS) algorithm for Level 1.
    - Ensure the agent can navigate the 2D environment successfully.
    - Handle scenarios where a path may not exist.
    - Integrate the algorithm seamlessly with the overall system.
2. Level 1 Solver DFS (Vo Hoang Phuc Khang):
    - Implement the Depth-First Search (DFS) algorithm for Level 1.
    - Provide a solution that explores paths deeply before backtracking.
    - Integrate the DFS algorithm into the navigation system.
3. Level 1 Solver UCS (Vuong Quoc Phong):
    - Develop the Uniform-Cost Search (UCS) algorithm for Level 1.
    - Focus on finding the path with the lowest cost.
    - Integrate the UCS algo seamlessly into the overall navigation system.
4. Level 1 Solver A Star (Hoang Thien Duc):

- Implement the A* algorithm for Level 1.
- Utilize heuristic search to improve the efficiency of pathfinding.
- Integrate the A* algorithm into the navigation system.

5. Level 2 Solver (Vo Hoang Phuc Khang):
   - Extend the solver to handle Level 2 scenarios, incorporating doors and keys.
   - Implement BFS for finding keys at the first time, and then BFS through the doors following the keys found in the last time to find other keys and repeat this process until reaching Mr. Thanh successfully or cannot reach any further grids. This process will also be associated with the key-finding order for every single grid.
   - Finally, for the key-finding order of Mr. Thanh grid, we use BFS to construct the following path.
6. Level 3 Solver (Nguyen Trinh Duy):
   - Develop the solver for Level 3, involving multiple floors and staircases.
   - The logic and path-finding process are strikingly similar to that of Level 2 Solver.
7. Level 4 Solver (Nguyen Trinh Duy):
   - Extend the solver to handle Level 4 scenarios with multiple agents.
   - For each turn, implement the pathfinding process like Level 3 Solver to determine the next step for each agent (from A1 to An, respectively), then keep repeating it.
   - If an agent which is not A1 steps on its destination, use random functions of Python Standard Library to generate a new destination as well as reset its path.
   - If A1 steps on its destination, the process terminates.
8. GUI Displayer for Algorithm (Vuong Quoc Phong):
   - Create a graphical user interface (GUI) to display the step-by-step process of the algorithms.
   - Ensure the GUI provides clear visualizations of the agent's movements.
9. Exporting Heatmap (Vuong Quoc Phong):
   - Implement functionality to export an image file showing a heatmap.
   - Differentiate colors for each agent, indicating the frequency of passing through each cell.
10. Generate Test Cases (Hoang Thien Duc):
    - Develop a set of diverse test cases to thoroughly evaluate the implemented algorithms.
    - Consider edge cases and scenarios that may pose challenges to the navigation system.
11. Demo Video (Nguyen Trinh Duy):
    - Create a demonstration video showcasing the functionality of the navigation system.

- Clearly explain and illustrate how each algorithm performs in different scenarios.

12. Report Documenting (All Team Members):
    - Collaborate on creating a comprehensive report documenting the project.
    - Provide insights into the performance of each algorithm in different levels and scenarios.

IV. Self-evaluation:

In summary, our team has managed to finish all the work on time and our team approach has successfully solved simple to medium difficulty test cases. However, during the testing phase, issues are found in level 4, and we must do extra work to fix it. Other edge cases should be tested more carefully.

V. Experiments:

Test case:

1. Level 1:
    - map0_L1.txt:
        - Size: 10x10 grid
        - Environment: Open terrain with walls, buildings, and the target Mr. Thanh.
        - Initial Position: Agent A1 starting at (1,0).
    - map1_L1.txt:
        - Size: 20x20 grid
        - Environment: Complex terrain with walls, open spaces, and Mr. Thanh located inside a building.
        - Initial Position: Agent A1 starting at (1,0).
    - map2_L1.txt:
        - Size: 20x20 grid
        - Environment: Labyrinthine structure with walls, doors, keys, and Mr. Thanh inside a room.
        - Initial Position: Agent A1 starting at (19,0).
    - map3_L1.txt:
        - Size: 19x19 grid
        - Environment: Complex terrain with walls and open spaces. Mr. Thanh is in a central position.
        - Initial Position: Agent A1 starting at (13,18).
    - map4_L1.txt:
        - Size: 20x20 grid

    - Environment: Diverse terrain with walls, open spaces, doors, and Mr. Thanh.
    - Initial Position: Agent A1 starting at (3,2).

All the test cases in Level 1 are easily solved by all the algorithms.

2. Level 2:

- map0_L2.txt:
    - Size: 10x14 grid
    - Environment: Moderate-sized grid with keys, doors, and Mr. Thanh.
    - Initial Position: Agent A1 starting at (1,1).
    - Comment: Introduction to keys and doors.
- map1_L2.txt:
    - Size: 16x16 grid
    - Environment: Larger grid with increased complexity.
    - Initial Position: Agent A1 starting at (1,0).
    - Comment: More keys and doors.
- map2_L2.txt:
    - Size: 20x20 grid
    - Environment: Larger grid with a centralized layout.
    - Initial Position: Agent A1 starting at (0,0).
    - Comment: Complex environment with diverse elements, but in this case, there is no solution.
- map3_L2.txt:
    - Size: 19x19 grid
    - Environment: Maze-like structure with various elements.
    - Initial Position: Agent A1 starting at (1,1).
    - Comment: Intricate maze layout with obstacles.
- map4_L2.txt:
    - Size: 20x20 grid
    - Environment: Diverse terrain with walls, doors, and Mr. Thanh.
    - Initial Position: Agent A1 starting at (3,2).
    - Comment: Offer a complex case for agent.

3. Level 3

- map0_L3.txt:
    - Size: 16x18 grid with 2 floors
    - Environment: Moderate-sized grid with keys, doors, stairs, and Mr. Thanh.

    - Initial Position: Agent A1 starting at (1,1) at floor 1
    - Comments: Simple case to introduce to the stairs
- map1_L3.txt:
    - Size: 20x20 grid with 2 floors
    - Environment: Large-sized grid with keys, doors, stairs, and Mr. Thanh.
    - Initial Position: Agent A1 starting at (0,9) at floor 1
    - Comment: More stairs and floors make the agent should travel between stairs more.
- map2_L3.txt:
    - Size: 19x19 grid with 3 floors
    - Environment: Large-sized grid with keys, doors, stairs, and Mr. Thanh.
    - Initial Position: Agent A1 starting at (1,1) at floor 1
    - Comment: More keys, doors, obstacles, and stairs make it more complex.
- map3_L3.txt:
    - Size: 20x20 grid with 2 floors
    - Environment: Large grid with a centralized layout.
    - Initial Position: Agent A1 starting at (0,0) at floor 1
    - Comment: T1 is in the center and the doors overlap to make it more complex.
- map4_L3.txt:
    - Size: 20x20 grid with 2 floors
    - Environment: Large-sized grid with keys, doors, stairs, and Mr. Thanh.
    - Initial Position: Agent A1 starting at (3,2) at floor 1
    - Comment: The keys are located separately, far away from each other and blocked by many layers of doors, making it more complex.

4. Level 4:
- map0_L4.txt:
    - Size: 16x18 grid with 2 floors
    - Comments:
        + The 3 sub-agents have paths that do not meet walls or doors.
        + Only A1 must find doors.
        + Exists 2 path of the sub-agents overlapped each other's.
- map1_L4.txt:
    - Size: 15x15 grid with 2 floors
    - Comments:
        + The road in this map is extremely narrow.

        - Because of the narrow road, the sub-agents blocked A1 and there is no solution
- map2_L4.txt:
    - Size: 19x19 grid with 3 floors
    - Comments:
        - Complex structures with a lot of obstacles and most of lanes are single lanes.
        - A1 blocks the sub-agents
- map3_L4.txt:
    - Size: 20x20 grid with 3 floors
    - Comments:
        - The map is a square maze with T1 in the center.
        - Sub-agents and their goals placed scattered outside the maze.
- map4_L4.txt:
    - Size: 20x20 grid with 2 floors
    - Comments:
        - The lanes being blocked by the doors a lot.
        - The goal of A1 place right next to a lot of goals of other sub-agents leads to stuck and makes this case have no solution.

Link for demo video: https://youtu.be/YRiNHQDESMM